

Harvard Lomax and Thomas H. Pulliam
NASA Ames Research Center

David W. Zingg
Institute for Aerospace Studies
University of Toronto

Fundamentals of Computational Fluid Dynamics

SPIN Springer's internal project number, if known

Physics – Monograph
(Editorial W. Beiglböck)

January 23, 2002

Springer-Verlag

Berlin Heidelberg New York
London Paris Tokyo
Hong Kong Barcelona
Budapest

To our families:
Joan, Harvard Laird, James, and Melinda
Carol, Teela, Troy, Merle Lynne, and Richie
Catherine, Natasha, Genevieve, and Trevor

Preface

The field of computational fluid dynamics (CFD) has already had a significant impact on the science and engineering of fluid dynamics, ranging from a role in aircraft design to enhancing our understanding of turbulent flows. It is thus not surprising that there exist several excellent books on the subject. We do not attempt to duplicate material which is thoroughly covered in these books. In particular, our book does not describe the most recent developments in algorithms, nor does it give any instruction with respect to programming. Neither turbulence modelling nor grid generation are covered. This book is intended for a reader who seeks a deep understanding of the fundamental principles which provide the foundation for the algorithms used in CFD. As a result of this focus, the book is suitable for a first course in CFD, presumably at the graduate level.

The underlying philosophy is that the theory of linear algebra and the attendant eigenanalysis of linear systems provide a mathematical framework to describe and unify most numerical methods in common use for solving the partial differential equations governing the physics of fluid flow. This approach originated with the first author during his long and distinguished career as Chief of the CFD Branch at the NASA Ames Research Center. He believed that a thorough understanding of sophisticated numerical algorithms should be as natural to a physicist as a thorough understanding of calculus. The material evolved from the notes for the course *Introduction to Computational Fluid Dynamics* taught by the first author at Stanford University from 1970 to 1994. The second author has taught the course since 1994, while the third author has taught a similar course at the University of Toronto since 1988.

The material is generally suitable for a typical graduate course of roughly fifteen weeks. It is unlikely that all of the book can be covered in the lectures and it is left to the instructor to set priorities. One approach which has worked well is to devote one lecture to each chapter, covering as much material as possible and leaving the remainder as reading material. The most essential material is in Chapters 2, 3, 4, 6, 7, and 8. Most of the chapters should be covered in the sequence used in the book. Exceptions include Chapter 5, which can be treated later in the course if desired, as well as Chapters 11 through 13, which can be covered anytime after Chapter 8. The mathematics

background associated with a typical undergraduate degree in engineering should be sufficient to understand the material.

The second and third authors would like to acknowledge their enormous debt to Harvard Lomax, both as a mentor and a friend. Sadly, Harvard passed away in May 1999. We present this book as further evidence of his outstanding legacy in CFD.

Moffett Field and Toronto,
August 2000

Thomas H. Pulliam
David W. Zingg

Contents

1. INTRODUCTION	1
1.1 Motivation	1
1.2 Background	2
1.2.1 Problem Specification and Geometry Preparation	2
1.2.2 Selection of Governing Equations and Boundary Conditions	3
1.2.3 Selection of Gridding Strategy and Numerical Method	3
1.2.4 Assessment and Interpretation of Results	4
1.3 Overview	4
1.4 Notation	4
2. CONSERVATION LAWS AND THE MODEL EQUATIONS	7
2.1 Conservation Laws	7
2.2 The Navier-Stokes and Euler Equations	8
2.3 The Linear Convection Equation	11
2.3.1 Differential Form	11
2.3.2 Solution in Wave Space	12
2.4 The Diffusion Equation	13
2.4.1 Differential Form	13
2.4.2 Solution in Wave Space	14
2.5 Linear Hyperbolic Systems	15
Exercises	17
3. FINITE-DIFFERENCE APPROXIMATIONS	19
3.1 Meshes and Finite-Difference Notation	19
3.2 Space Derivative Approximations	21
3.3 Finite-Difference Operators	22
3.3.1 Point Difference Operators	22
3.3.2 Matrix Difference Operators	23
3.3.3 Periodic Matrices	26
3.3.4 Circulant Matrices	27
3.4 Constructing Differencing Schemes of Any Order	28
3.4.1 Taylor Tables	28
3.4.2 Generalization of Difference Formulas	31

3.4.3	Lagrange and Hermite Interpolation Polynomials	33
3.4.4	Practical Application of Padé Formulas	34
3.4.5	Other Higher-Order Schemes	36
3.5	Fourier Error Analysis	37
3.5.1	Application to a Spatial Operator	37
3.6	Difference Operators at Boundaries	41
3.6.1	The Linear Convection Equation	41
3.6.2	The Diffusion Equation	43
	Exercises	46

1. INTRODUCTION

1.1 Motivation

The material in this book originated from attempts to understand and systemize numerical solution techniques for the partial differential equations governing the physics of fluid flow. As time went on and these attempts began to crystallize, underlying constraints on the nature of the material began to form. The principal such constraint was the demand for unification. Was there one mathematical structure which could be used to describe the behavior and results of most numerical methods in common use in the field of fluid dynamics? Perhaps the answer is arguable, but the authors believe the answer is affirmative and present this book as justification for that belief. The mathematical structure is the theory of linear algebra and the attendant eigenanalysis of linear systems.

The ultimate goal of the field of computational fluid dynamics (CFD) is to understand the physical events that occur in the flow of fluids around and within designated objects. These events are related to the action and interaction of phenomena such as dissipation, diffusion, convection, shock waves, slip surfaces, boundary layers, and turbulence. In the field of aerodynamics, all of these phenomena are governed by the compressible Navier-Stokes equations. Many of the most important aspects of these relations are nonlinear and, as a consequence, often have no analytic solution. This, of course, motivates the numerical solution of the associated partial differential equations. At the same time it would seem to invalidate the use of linear algebra for the classification of the numerical methods. Experience has shown that such is not the case.

As we shall see in a later chapter, the use of numerical methods to solve partial differential equations introduces an approximation that, in effect, can change the form of the basic partial differential equations themselves. The new equations, which are the ones actually being solved by the numerical process, are often referred to as the modified partial differential equations. Since they are not precisely the same as the original equations, they can, and probably will, simulate the physical phenomena listed above in ways that are not exactly the same as an exact solution to the basic partial differential equation. Mathematically, these differences are usually referred to as truncation errors. However, the theory associated with the numerical analysis of

fluid mechanics was developed predominantly by scientists deeply interested in the physics of fluid flow and, as a consequence, these errors are often identified with a particular physical phenomenon on which they have a strong effect. Thus methods are said to have a lot of “artificial viscosity” or said to be highly dispersive. This means that the errors caused by the numerical approximation result in a modified partial differential equation having additional terms that can be identified with the physics of dissipation in the first case and dispersion in the second. There is nothing wrong, of course, with identifying an error with a physical process, nor with deliberately directing an error to a specific physical process, as long as the error remains in some engineering sense “small”. It is safe to say, for example, that most numerical methods in practical use for solving the nondissipative Euler equations create a modified partial differential equation that produces some form of dissipation. However, if used and interpreted properly, these methods give very useful information.

Regardless of what the numerical errors are called, if their effects are not thoroughly understood and controlled, they can lead to serious difficulties, producing answers that represent little, if any, physical reality. This motivates studying the concepts of stability, convergence, and consistency. On the other hand, even if the errors are kept small enough that they can be neglected (for engineering purposes), the resulting simulation can still be of little practical use if inefficient or inappropriate algorithms are used. This motivates studying the concepts of stiffness, factorization, and algorithm development in general. All of these concepts we hope to clarify in this book.

1.2 Background

The field of computational fluid dynamics has a broad range of applicability. Independent of the specific application under study, the following sequence of steps generally must be followed in order to obtain a satisfactory solution.

1.2.1 Problem Specification and Geometry Preparation

The first step involves the specification of the problem, including the geometry, flow conditions, and the requirements of the simulation. The geometry may result from measurements of an existing configuration or may be associated with a design study. Alternatively, in a design context, no geometry need be supplied. Instead, a set of objectives and constraints must be specified. Flow conditions might include, for example, the Reynolds number and Mach number for the flow over an airfoil. The requirements of the simulation include issues such as the level of accuracy needed, the turnaround time required, and the solution parameters of interest. The first two of these requirements are often in conflict and compromise is necessary. As an example

of solution parameters of interest in computing the flowfield about an airfoil, one may be interested in i) the lift and pitching moment only, ii) the drag as well as the lift and pitching moment, or iii) the details of the flow at some specific location.

1.2.2 Selection of Governing Equations and Boundary Conditions

Once the problem has been specified, an appropriate set of governing equations and boundary conditions must be selected. It is generally accepted that the phenomena of importance to the field of continuum fluid dynamics are governed by the conservation of mass, momentum, and energy. The partial differential equations resulting from these conservation laws are referred to as the Navier-Stokes equations. However, in the interest of efficiency, it is always prudent to consider solving simplified forms of the Navier-Stokes equations when the simplifications retain the physics which are essential to the goals of the simulation. Possible simplified governing equations include the potential-flow equations, the Euler equations, and the thin-layer Navier-Stokes equations. These may be steady or unsteady and compressible or incompressible. Boundary types which may be encountered include solid walls, inflow and outflow boundaries, periodic boundaries, symmetry boundaries, etc. The boundary conditions which must be specified depend upon the governing equations. For example, at a solid wall, the Euler equations require flow tangency to be enforced, while the Navier-Stokes equations require the no-slip condition. If necessary, physical models must be chosen for processes which cannot be simulated within the specified constraints. Turbulence is an example of a physical process which is rarely simulated in a practical context (at the time of writing) and thus is often modeled. The success of a simulation depends greatly on the engineering insight involved in selecting the governing equations and physical models based on the problem specification.

1.2.3 Selection of Gridding Strategy and Numerical Method

Next a numerical method and a strategy for dividing the flow domain into cells, or elements, must be selected. We concern ourselves here only with numerical methods requiring such a tessellation of the domain, which is known as a grid, or mesh. Many different gridding strategies exist, including structured, unstructured, hybrid, composite, and overlapping grids. Furthermore, the grid can be altered based on the solution in an approach known as solution-adaptive gridding. The numerical methods generally used in CFD can be classified as finite-difference, finite-volume, finite-element, or spectral methods. The choices of a numerical method and a gridding strategy are strongly interdependent. For example, the use of finite-difference methods is typically restricted to structured grids. Here again, the success of a simulation can depend on appropriate choices for the problem or class of problems of interest.

1.2.4 Assessment and Interpretation of Results

Finally, the results of the simulation must be assessed and interpreted. This step can require post-processing of the data, for example calculation of forces and moments, and can be aided by sophisticated flow visualization tools and error estimation techniques. It is critical that the magnitude of both numerical and physical-model errors be well understood.

1.3 Overview

It should be clear that successful simulation of fluid flows can involve a wide range of issues from grid generation to turbulence modelling to the applicability of various simplified forms of the Navier-Stokes equations. Many of these issues are not addressed in this book. Instead we focus on numerical methods, with emphasis on finite-difference and finite-volume methods for the Euler and Navier-Stokes equations. Rather than presenting the details of the most advanced methods, which are still evolving, we present a foundation for developing, analyzing, and understanding such methods.

Fortunately, to develop, analyze, and understand most numerical methods used to find solutions for the complete compressible Navier-Stokes equations, we can make use of much simpler expressions, the so-called “model” equations. These model equations isolate certain aspects of the physics contained in the complete set of equations. Hence their numerical solution can illustrate the properties of a given numerical method when applied to a more complicated system of equations which governs similar physical phenomena. Although the model equations are extremely simple and easy to solve, they have been carefully selected to be representative, when used intelligently, of difficulties and complexities that arise in realistic two- and three-dimensional fluid flow simulations. We believe that a thorough understanding of what happens when numerical approximations are applied to the model equations is a major first step in making confident and competent use of numerical approximations to the Euler and Navier-Stokes equations. As a word of caution, however, it should be noted that, although we can learn a great deal by studying numerical methods as applied to the model equations and can use that information in the design and application of numerical methods to practical problems, there are many aspects of practical problems which can only be understood in the context of the complete physical systems.

1.4 Notation

The notation is generally explained as it is introduced. Bold type is reserved for real physical vectors, such as velocity. The vector symbol $\vec{\tau}$ is used for the vectors (or column matrices) which contain the values of the dependent

variable at the nodes of a grid. Otherwise, the use of a vector consisting of a collection of scalars should be apparent from the context and is not identified by any special notation. For example, the variable u can denote a scalar Cartesian velocity component in the Euler and Navier-Stokes equations, a scalar quantity in the linear convection and diffusion equations, and a vector consisting of a collection of scalars in our presentation of hyperbolic systems. Some of the abbreviations used throughout the text are listed and defined below.

PDE	Partial differential equation
ODE	Ordinary differential equation
O Δ E	Ordinary difference equation
RHS	Right-hand side
P.S.	Particular solution of an ODE or system of ODE's
S.S.	Fixed (time-invariant) steady-state solution
k-D	k-dimensional space
(\vec{bc})	Boundary conditions, usually a vector
O(α)	A term of order (i.e., proportional to) α
$\Re(z)$	The real part of a complex number, z

2. CONSERVATION LAWS AND THE MODEL EQUATIONS

We start out by casting our equations in the most general form, the integral conservation-law form, which is useful in understanding the concepts involved in finite-volume schemes. The equations are then recast into divergence form, which is natural for finite-difference schemes. The Euler and Navier-Stokes equations are briefly discussed in this chapter. The main focus, though, will be on representative model equations, in particular, the convection and diffusion equations. These equations contain many of the salient mathematical and physical features of the full Navier-Stokes equations. The concepts of convection and diffusion are prevalent in our development of numerical methods for computational fluid dynamics, and the recurring use of these model equations allows us to develop a consistent framework of analysis for consistency, accuracy, stability, and convergence. The model equations we study have two properties in common. They are linear partial differential equations (PDE's) with coefficients that are constant in both space and time, and they represent phenomena of importance to the analysis of certain aspects of fluid dynamics problems.

2.1 Conservation Laws

Conservation laws, such as the Euler and Navier-Stokes equations and our model equations, can be written in the following integral form:

$$\int_{V(t_2)} Q dV - \int_{V(t_1)} Q dV + \int_{t_1}^{t_2} \oint_{S(t)} \mathbf{n} \cdot \mathbf{F} dS dt = \int_{t_1}^{t_2} \int_{V(t)} P dV dt \quad (2.1)$$

In this equation, Q is a vector containing the set of variables which are conserved, e.g., mass, momentum, and energy, per unit volume. The equation is a statement of the conservation of these quantities in a finite region of space with volume $V(t)$ and surface area $S(t)$ over a finite interval of time $t_2 - t_1$. In two dimensions, the region of space, or cell, is an area $A(t)$ bounded by a closed contour $C(t)$. The vector \mathbf{n} is a unit vector normal to the surface pointing outward, \mathbf{F} is a set of vectors, or tensor, containing the flux of Q per unit area per unit time, and P is the rate of production of Q per unit volume per unit time. If all variables are continuous in time, then Eq. 2.1 can be rewritten as

$$\frac{d}{dt} \int_{V(t)} Q dV + \oint_{S(t)} \mathbf{n} \cdot \mathbf{F} dS = \int_{V(t)} P dV \quad (2.2)$$

Those methods which make various numerical approximations of the integrals in Eqs. 2.1 and 2.2 and find a solution for Q on that basis are referred to as *finite-volume methods*. Many of the advanced codes written for CFD applications are based on the finite-volume concept.

On the other hand, a partial derivative form of a conservation law can also be derived. The divergence form of Eq. 2.2 is obtained by applying Gauss's theorem to the flux integral, leading to

$$\frac{\partial Q}{\partial t} + \nabla \cdot \mathbf{F} = P \quad (2.3)$$

where $\nabla \cdot$ is the well-known divergence operator given, in Cartesian coordinates, by

$$\nabla \cdot \equiv \left(\mathbf{i} \frac{\partial}{\partial x} + \mathbf{j} \frac{\partial}{\partial y} + \mathbf{k} \frac{\partial}{\partial z} \right). \quad (2.4)$$

and \mathbf{i} , \mathbf{j} , and \mathbf{k} are unit vectors in the x , y , and z coordinate directions, respectively. Those methods which make various approximations of the derivatives in Eq. 2.3 and find a solution for Q on that basis are referred to as *finite-difference methods*.

2.2 The Navier-Stokes and Euler Equations

The Navier-Stokes equations form a coupled system of nonlinear PDE's describing the conservation of mass, momentum and energy for a fluid. For a Newtonian fluid in one dimension, they can be written as

$$\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial x} = 0 \quad (2.5)$$

with

$$Q = \begin{bmatrix} \rho \\ \rho u \\ e \end{bmatrix}, \quad E = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ u(e + p) \end{bmatrix} - \begin{bmatrix} 0 \\ \frac{4}{3}\mu \frac{\partial u}{\partial x} \\ \frac{4}{3}\mu u \frac{\partial u}{\partial x} + \kappa \frac{\partial T}{\partial x} \end{bmatrix} \quad (2.6)$$

where ρ is the fluid density, u is the velocity, e is the total energy per unit volume, p is the pressure, T is the temperature, μ is the coefficient of viscosity, and κ is the thermal conductivity. The total energy e includes internal energy per unit volume $\rho\epsilon$ (where ϵ is the internal energy per unit mass) and kinetic energy per unit volume $\rho u^2/2$. These equations must be supplemented by relations between μ and κ and the fluid state as well as an equation of state, such as the ideal gas law. Details can be found in many textbooks. Note that the convective fluxes lead to first derivatives in space, while the viscous and heat conduction terms involve second derivatives. This form of the equations

is called *conservation-law* or *conservative* form. Non-conservative forms can be obtained by expanding derivatives of products using the product rule or by introducing different dependent variables, such as u and p . Although non-conservative forms of the equations are *analytically* the same as the above form, they can lead to quite different *numerical* solutions in terms of shock strength and shock speed, for example. Thus the conservative form is appropriate for solving flows with features such as shock waves.

Many flows of engineering interest are steady (time-invariant), or at least may be treated as such. For such flows, we are often interested in the steady-state solution of the Navier-Stokes equations, with no interest in the transient portion of the solution. The steady solution to the one-dimensional Navier-Stokes equations must satisfy

$$\frac{\partial E}{\partial x} = 0 \quad (2.7)$$

If we neglect viscosity and heat conduction, the Euler equations are obtained. In two-dimensional Cartesian coordinates, these can be written as

$$\frac{\partial Q}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} = 0 \quad (2.8)$$

with

$$Q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix}, \quad E = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(e + p) \end{bmatrix}, \quad F = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(e + p) \end{bmatrix} \quad (2.9)$$

where u and v are the Cartesian velocity components. Later on we will make use of the following form of the Euler equations as well:

$$\frac{\partial Q}{\partial t} + A \frac{\partial Q}{\partial x} + B \frac{\partial Q}{\partial y} = 0 \quad (2.10)$$

The matrices $A = \frac{\partial E}{\partial Q}$ and $B = \frac{\partial F}{\partial Q}$ are known as the flux Jacobians. The flux vectors given above are written in terms of the *primitive* variables, ρ , u , v , and p . In order to derive the flux Jacobian matrices, we must first write the flux vectors E and F in terms of the *conservative* variables, q_1 , q_2 , q_3 , and q_4 , as follows:

$$E = \begin{bmatrix} E_1 \\ E_2 \\ E_3 \\ E_4 \end{bmatrix} = \begin{bmatrix} q_2 \\ (\gamma - 1)q_4 + \frac{3 - \gamma}{2} \frac{q_2^2}{q_1} - \frac{\gamma - 1}{2} \frac{q_3^2}{q_1} \\ \frac{q_3 q_2}{q_1} \\ \gamma \frac{q_4 q_2}{q_1} - \frac{\gamma - 1}{2} \left(\frac{q_2^3}{q_1^2} + \frac{q_3^2 q_2}{q_1^2} \right) \end{bmatrix} \quad (2.11)$$

$$F = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} = \begin{bmatrix} q_3 \\ \frac{q_3 q_2}{q_1} \\ (\gamma - 1)q_4 + \frac{3 - \gamma}{2} \frac{q_3^2}{q_1} - \frac{\gamma - 1}{2} \frac{q_2^2}{q_1} \\ \gamma \frac{q_4 q_3}{q_1} - \frac{\gamma - 1}{2} \left(\frac{q_2^2 q_3}{q_1} + \frac{q_3^3}{q_1} \right) \end{bmatrix} \quad (2.12)$$

We have assumed that the pressure satisfies $p = (\gamma - 1)[e - \rho(u^2 + v^2)/2]$ from the ideal gas law, where γ is the ratio of specific heats, c_p/c_v . From this it follows that the flux Jacobian of E can be written in terms of the conservative variables as

$$A = \frac{\partial E_i}{\partial q_j} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a_{21} & (3 - \gamma)\left(\frac{q_2}{q_1}\right) & (1 - \gamma)\left(\frac{q_3}{q_1}\right) & \gamma - 1 \\ -\left(\frac{q_2}{q_1}\right)\left(\frac{q_3}{q_1}\right) & \left(\frac{q_3}{q_1}\right) & \left(\frac{q_2}{q_1}\right) & 0 \\ a_{41} & a_{42} & a_{43} & \gamma\left(\frac{q_2}{q_1}\right) \end{bmatrix} \quad (2.13)$$

where

$$\begin{aligned} a_{21} &= \frac{\gamma - 1}{2} \left(\frac{q_3}{q_1}\right)^2 - \frac{3 - \gamma}{2} \left(\frac{q_2}{q_1}\right)^2 \\ a_{41} &= (\gamma - 1) \left[\left(\frac{q_2}{q_1}\right)^3 + \left(\frac{q_3}{q_1}\right)^2 \left(\frac{q_2}{q_1}\right) \right] - \gamma \left(\frac{q_4}{q_1}\right) \left(\frac{q_2}{q_1}\right) \\ a_{42} &= \gamma \left(\frac{q_4}{q_1}\right) - \frac{\gamma - 1}{2} \left[3 \left(\frac{q_2}{q_1}\right)^2 + \left(\frac{q_3}{q_1}\right)^2 \right] \\ a_{43} &= -(\gamma - 1) \left(\frac{q_2}{q_1}\right) \left(\frac{q_3}{q_1}\right) \end{aligned} \quad (2.14)$$

and in terms of the primitive variables as

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a_{21} & (3-\gamma)u & (1-\gamma)v & (\gamma-1) \\ -uv & v & u & 0 \\ a_{41} & a_{42} & a_{43} & \gamma u \end{bmatrix} \quad (2.15)$$

where

$$\begin{aligned} a_{21} &= \frac{\gamma-1}{2}v^2 - \frac{3-\gamma}{2}u^2 \\ a_{41} &= (\gamma-1)u(u^2+v^2) - \gamma\frac{ue}{\rho} \\ a_{42} &= \gamma\frac{e}{\rho} - \frac{\gamma-1}{2}(3u^2+v^2) \\ a_{43} &= (1-\gamma)uv \end{aligned} \quad (2.16)$$

Derivation of the two forms of $B = \partial F / \partial Q$ is similar. The eigenvalues of the flux Jacobian matrices are purely real. This is the defining feature of *hyperbolic* systems of PDE's, which are further discussed in Section 2.5. The homogeneous property of the Euler equations is discussed in Appendix ??.

The Navier-Stokes equations include both convective and diffusive fluxes. This motivates the choice of our two scalar model equations associated with the physics of convection and diffusion. Furthermore, aspects of convective phenomena associated with coupled systems of equations such as the Euler equations are important in developing numerical methods and boundary conditions. Thus we also study linear hyperbolic systems of PDE's.

2.3 The Linear Convection Equation

2.3.1 Differential Form

The simplest linear model for convection and wave propagation is the linear convection equation given by the following PDE:

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0 \quad (2.17)$$

Here $u(x, t)$ is a scalar quantity propagating with speed a , a real constant which may be positive or negative. The manner in which the boundary conditions are specified separates the following two phenomena for which this equation is a model:

- (1) In one type, the scalar quantity u is given on one boundary, corresponding to a wave entering the domain through this “inflow” boundary. No boundary condition is specified at the opposite side, the “outflow” boundary. This is consistent in terms of the well-posedness of a first-order PDE. Hence the wave leaves the domain through the outflow boundary without distortion or reflection. This type of phenomenon is referred to, simply, as the convection problem. It represents most of the “usual” situations encountered in convecting systems. Note that the left-hand boundary is the inflow boundary when a is positive, while the right-hand boundary is the inflow boundary when a is negative.
- (2) In the other type, the flow being simulated is periodic. At any given time, what enters on one side of the domain must be the same as that which is leaving on the other. This is referred to as the *biconvection* problem. It is the simplest to study and serves to illustrate many of the basic properties of numerical methods applied to problems involving convection, without special consideration of boundaries. Hence, we pay a great deal of attention to it in the initial chapters.

Now let us consider a situation in which the initial condition is given by $u(x, 0) = u_0(x)$, and the domain is infinite. It is easy to show by substitution that the exact solution to the linear convection equation is then

$$u(x, t) = u_0(x - at) \quad (2.18)$$

The initial waveform propagates unaltered with speed $|a|$ to the right if a is positive, and to the left if a is negative. With periodic boundary conditions, the waveform travels through one boundary and reappears at the other boundary, eventually returning to its initial position. In this case, the process continues forever without any change in the shape of the solution. Preserving the shape of the initial condition $u_0(x)$ can be a difficult challenge for a numerical method.

2.3.2 Solution in Wave Space

We now examine the biconvection problem in more detail. Let the domain be given by $0 \leq x \leq 2\pi$. We restrict our attention to initial conditions in the form

$$u(x, 0) = f(0)e^{i\kappa x} \quad (2.19)$$

where $f(0)$ is a complex constant, and κ is the *wavenumber*. In order to satisfy the periodic boundary conditions, κ must be an integer. It is a measure of the number of wavelengths within the domain. With such an initial condition, the solution can be written as

$$u(x, t) = f(t)e^{i\kappa x} \quad (2.20)$$

where the time dependence is contained in the complex function $f(t)$. Substituting this solution into the linear convection equation, Eq. 2.17, we find that $f(t)$ satisfies the following ordinary differential equation (ODE)

$$\frac{df}{dt} = -ia\kappa f \quad (2.21)$$

which has the solution

$$f(t) = f(0)e^{-ia\kappa t} \quad (2.22)$$

Substituting $f(t)$ into Eq. 2.20 gives the following solution

$$u(x, t) = f(0)e^{i\kappa(x-at)} = f(0)e^{i(\kappa x - \omega t)} \quad (2.23)$$

where the frequency, ω , the wavenumber, κ , and the phase speed, a , are related by

$$\omega = \kappa a \quad (2.24)$$

The relation between the frequency and the wavenumber is known as the dispersion relation. The linear relation given by Eq. 2.24 is characteristic of wave propagation in a nondispersive medium. This means that the phase speed is the same for all wavenumbers. As we shall see later, most numerical methods introduce some dispersion; that is, in a simulation, waves with different wavenumbers travel at different speeds.

An arbitrary initial waveform can be produced by summing initial conditions of the form of Eq. 2.19. For M modes, one obtains

$$u(x, 0) = \sum_{m=1}^M f_m(0)e^{i\kappa_m x} \quad (2.25)$$

where the wavenumbers are often ordered such that $\kappa_1 \leq \kappa_2 \leq \dots \leq \kappa_M$. Since the wave equation is linear, the solution is obtained by summing solutions of the form of Eq. 2.23, giving

$$u(x, t) = \sum_{m=1}^M f_m(0)e^{i\kappa_m(x-at)} \quad (2.26)$$

Dispersion and dissipation resulting from a numerical approximation will cause the shape of the solution to change from that of the original waveform.

2.4 The Diffusion Equation

2.4.1 Differential Form

Diffusive fluxes are associated with molecular motion in a continuum fluid. A simple linear model equation for a diffusive process is

$$\frac{\partial u}{\partial t} = \nu \frac{\partial^2 u}{\partial x^2} \quad (2.27)$$

where ν is a positive real constant. For example, with u representing the temperature, this parabolic PDE governs the diffusion of heat in one dimension. Boundary conditions can be periodic, Dirichlet (specified u), Neumann (specified $\partial u/\partial x$), or mixed Dirichlet/Neumann.

In contrast to the linear convection equation, the diffusion equation has a nontrivial *steady-state* solution, which is one that satisfies the governing PDE with the partial derivative in time equal to zero. In the case of Eq. 2.27, the steady-state solution must satisfy

$$\frac{\partial^2 u}{\partial x^2} = 0 \quad (2.28)$$

Therefore, u must vary linearly with x at steady state such that the boundary conditions are satisfied. Other steady-state solutions are obtained if a source term $g(x)$ is added to Eq. 2.27, as follows:

$$\frac{\partial u}{\partial t} = \nu \left[\frac{\partial^2 u}{\partial x^2} - g(x) \right] \quad (2.29)$$

giving a steady state-solution which satisfies

$$\frac{\partial^2 u}{\partial x^2} - g(x) = 0 \quad (2.30)$$

In two dimensions, the diffusion equation becomes

$$\frac{\partial u}{\partial t} = \nu \left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} - g(x, y) \right] \quad (2.31)$$

where $g(x, y)$ is again a source term. The corresponding steady equation is

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} - g(x, y) = 0 \quad (2.32)$$

While Eq. 2.31 is parabolic, Eq. 2.32 is elliptic. The latter is known as the Poisson equation for nonzero g , and as Laplace's equation for zero g .

2.4.2 Solution in Wave Space

We now consider a series solution to Eq. 2.27 with Dirichlet boundary conditions. Let the domain be given by $0 \leq x \leq \pi$ with boundary conditions $u(0) = u_a$, $u(\pi) = u_b$. It is clear that the steady-state solution is given by a linear function which satisfies the boundary conditions, i.e., $h(x) = u_a + (u_b - u_a)x/\pi$. Let the initial condition be

$$u(x, 0) = \sum_{m=1}^M f_m(0) \sin \kappa_m x + h(x) \quad (2.33)$$

where κ must be an integer in order to satisfy the boundary conditions. A solution of the form

$$u(x, t) = \sum_{m=1}^M f_m(t) \sin \kappa_m x + h(x) \quad (2.34)$$

satisfies the initial and boundary conditions. Substituting this form into Eq. 2.27 gives the following ODE for f_m :

$$\frac{df_m}{dt} = -\kappa_m^2 \nu f_m \quad (2.35)$$

and we find

$$f_m(t) = f_m(0) e^{-\kappa_m^2 \nu t} \quad (2.36)$$

Substituting $f_m(t)$ into equation 2.34, we obtain

$$u(x, t) = \sum_{m=1}^M f_m(0) e^{-\kappa_m^2 \nu t} \sin \kappa_m x + h(x) \quad (2.37)$$

The steady-state solution ($t \rightarrow \infty$) is simply $h(x)$. Eq. 2.37 shows that high wavenumber components (large κ_m) of the solution decay more rapidly than low wavenumber components, consistent with the physics of diffusion.

2.5 Linear Hyperbolic Systems

The Euler equations, Eq. 2.8, form a hyperbolic system of partial differential equations. Other systems of equations governing convection and wave propagation phenomena, such as the Maxwell equations describing the propagation of electromagnetic waves, are also of hyperbolic type. Many aspects of numerical methods for such systems can be understood by studying a one-dimensional constant-coefficient linear system of the form

$$\frac{\partial u}{\partial t} + A \frac{\partial u}{\partial x} = 0 \quad (2.38)$$

where $u = u(x, t)$ is a vector of length m and A is a real $m \times m$ matrix. For conservation laws, this equation can also be written in the form

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} = 0 \quad (2.39)$$

where f is the flux vector and $A = \frac{\partial f}{\partial u}$ is the flux Jacobian matrix. The entries in the flux Jacobian are

$$a_{ij} = \frac{\partial f_i}{\partial u_j} \quad (2.40)$$

The flux Jacobian for the Euler equations is derived in Section 2.2.

Such a system is hyperbolic if A is diagonalizable with real eigenvalues.¹ Thus

$$\Lambda = X^{-1}AX \quad (2.41)$$

where Λ is a diagonal matrix containing the eigenvalues of A , and X is the matrix of right eigenvectors. Premultiplying Eq. 2.38 by X^{-1} , postmultiplying A by the product XX^{-1} , and noting that X and X^{-1} are constants, we obtain

$$\frac{\partial X^{-1}u}{\partial t} + \frac{\partial \overbrace{X^{-1}AX}^{\Lambda} X^{-1}u}{\partial x} = 0 \quad (2.42)$$

With $w = X^{-1}u$, this can be rewritten as

$$\frac{\partial w}{\partial t} + \Lambda \frac{\partial w}{\partial x} = 0 \quad (2.43)$$

When written in this manner, the equations have been decoupled into m scalar equations of the form

$$\frac{\partial w_i}{\partial t} + \lambda_i \frac{\partial w_i}{\partial x} = 0 \quad (2.44)$$

The elements of w are known as *characteristic variables*. Each characteristic variable satisfies the linear convection equation with the speed given by the corresponding eigenvalue of A .

Based on the above, we see that a hyperbolic system in the form of Eq. 2.38 has a solution given by the superposition of waves which can travel in either the positive or negative directions and at varying speeds. While the scalar linear convection equation is clearly an excellent model equation for hyperbolic systems, we must ensure that our numerical methods are appropriate for wave speeds of arbitrary sign and possibly widely varying magnitudes.

The one-dimensional Euler equations can also be diagonalized, leading to three equations in the form of the linear convection equation, although they remain nonlinear, of course. The eigenvalues of the flux Jacobian matrix, or wave speeds, are u , $u + c$, and $u - c$, where u is the local fluid velocity, and $c = \sqrt{\gamma p / \rho}$ is the local speed of sound. The speed u is associated with convection of the fluid, while $u + c$ and $u - c$ are associated with sound waves. Therefore, in a supersonic flow, where $|u| > c$, all of the wave speeds have the same sign. In a subsonic flow, where $|u| < c$, wave speeds of both positive and negative sign are present, corresponding to the fact that sound waves can travel upstream in a subsonic flow.

¹ See Appendix ?? for a brief review of some basic relations and definitions from linear algebra.

The signs of the eigenvalues of the matrix A are also important in determining suitable boundary conditions. The characteristic variables each satisfy the linear convection equation with the wave speed given by the corresponding eigenvalue. Therefore, the boundary conditions can be specified accordingly. That is, characteristic variables associated with positive eigenvalues can be specified at the left boundary, which corresponds to inflow for these variables. Characteristic variables associated with negative eigenvalues can be specified at the right boundary, which is the inflow boundary for these variables. While other boundary condition treatments are possible, they must be consistent with this approach.

Exercises

2.1 Show that the 1-D Euler equations can be written in terms of the *primitive* variables $R = [\rho, u, p]^T$ as follows:

$$\frac{\partial R}{\partial t} + M \frac{\partial R}{\partial x} = 0$$

where

$$M = \begin{bmatrix} u & \rho & 0 \\ 0 & u & \rho^{-1} \\ 0 & \gamma p & u \end{bmatrix}$$

Assume an ideal gas, $p = (\gamma - 1)(e - \rho u^2/2)$.

2.2 Find the eigenvalues and eigenvectors of the matrix M derived in question 2.1.

2.3 Derive the flux Jacobian matrix $A = \partial E / \partial Q$ for the 1-D Euler equations resulting from the conservative variable formulation (Eq. 2.5). Find its eigenvalues and compare with those obtained in question 2.2.

2.4 Show that the two matrices M and A derived in questions 2.1 and 2.3, respectively, are related by a similarity transform. (Hint: make use of the matrix $S = \partial Q / \partial R$.)

2.5 Write the 2-D diffusion equation, Eq. 2.31, in the form of Eq. 2.2.

2.6 Given the initial condition $u(x, 0) = \sin x$ defined on $0 \leq x \leq 2\pi$, write it in the form of Eq. 2.25, that is, find the necessary values of $f_m(0)$. (Hint: use $M = 2$ with $\kappa_1 = 1$ and $\kappa_2 = -1$.) Next consider the same initial condition defined only at $x = 2\pi j/4$, $j = 0, 1, 2, 3$. Find the values of $f_m(0)$ required to reproduce the initial condition at these discrete points using $M = 4$ with $\kappa_m = m - 1$.

2.7 Plot the first three basis functions used in constructing the exact solution to the diffusion equation in Section 2.4.2. Next consider a solution with boundary conditions $u_a = u_b = 0$, and initial conditions from Eq. 2.33 with $f_m(0) = 1$ for $1 \leq m \leq 3$, $f_m(0) = 0$ for $m > 3$. Plot the initial condition on the domain $0 \leq x \leq \pi$. Plot the solution at $t = 1$ with $\nu = 1$.

2.8 Write the classical wave equation $\partial^2 u / \partial t^2 = c^2 \partial^2 u / \partial x^2$ as a first-order system, i.e., in the form

$$\frac{\partial U}{\partial t} + A \frac{\partial U}{\partial x} = 0$$

where $U = [\partial u / \partial x, \partial u / \partial t]^T$. Find the eigenvalues and eigenvectors of A .

2.9 The Cauchy-Riemann equations are formed from the coupling of the steady compressible continuity (conservation of mass) equation

$$\frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} = 0$$

and the vorticity definition

$$\omega = -\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} = 0$$

where $\omega = 0$ for irrotational flow. For isentropic and homenthalpic flow, the system is closed by the relation

$$\rho = \left(1 - \frac{\gamma - 1}{2} (u^2 + v^2 - 1)\right)^{\frac{1}{\gamma - 1}}$$

Note that the variables have been nondimensionalized. Combining the two PDE's, we have

$$\frac{\partial f(q)}{\partial x} + \frac{\partial g(q)}{\partial y} = 0$$

where

$$q = \begin{pmatrix} u \\ v \end{pmatrix}, \quad f = \begin{pmatrix} -\rho u \\ v \end{pmatrix}, \quad g = \begin{pmatrix} -\rho v \\ -u \end{pmatrix}$$

One approach to solving these equations is to add a time-dependent term and find the steady solution of the following equation:

$$\frac{\partial q}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} = 0$$

- Find the flux Jacobians of f and g with respect to q .
- Determine the eigenvalues of the flux Jacobians.
- Determine the conditions (in terms of ρ and u) under which the system is hyperbolic, i.e., has real eigenvalues.
- Are the above fluxes homogeneous? (See Appendix ??.)

3. FINITE-DIFFERENCE APPROXIMATIONS

In common with the equations governing unsteady fluid flow, our model equations contain partial derivatives with respect to both space and time. One can approximate these simultaneously and then solve the resulting difference equations. Alternatively, one can approximate the spatial derivatives first, thereby producing a system of ordinary differential equations. The time derivatives are approximated next, leading to a *time-marching method* which produces a set of difference equations. This is the approach emphasized here. In this chapter, the concept of finite-difference approximations to partial derivatives is presented. While these can be applied either to spatial derivatives or time derivatives, our emphasis in this chapter is on spatial derivatives; time derivatives are treated in Chapter ???. Strategies for applying these finite-difference approximations will be discussed in Chapter ???.

All of the material below is presented in a Cartesian system. We emphasize the fact that quite general classes of meshes expressed in general curvilinear coordinates in *physical space* can be transformed to a uniform Cartesian mesh with equispaced intervals in a so-called *computational space*, as shown in Figure 3.1. The *computational space* is uniform; all the geometric variation is absorbed into variable coefficients of the transformed equations. For this reason, in much of the following accuracy analysis, we use an equispaced Cartesian system without being unduly restrictive or losing practical application.

3.1 Meshes and Finite-Difference Notation

The simplest mesh involving both time and space is shown in Figure 3.2. Inspection of this figure permits us to define the terms and notation needed to describe finite-difference approximations. In general, the dependent variables, u , for example, are functions of the independent variables t , and x, y, z . For the first several chapters we consider primarily the 1-D case $u = u(x, t)$. When only one variable is denoted, dependence on the other is assumed. The mesh index for x is *always* j , and that for t is *always* n . Then on an equispaced grid

$$x = x_j = j \Delta x \tag{3.1}$$

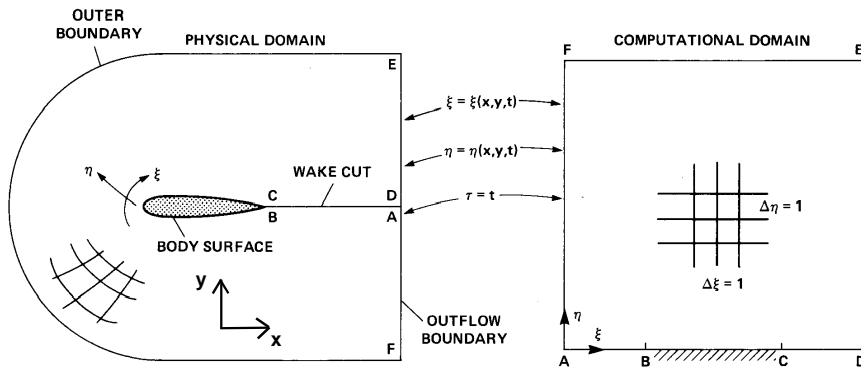


Fig. 3.1. Physical and computational spaces.

$$t = t_n = n\Delta t = nh \quad (3.2)$$

where Δx is the spacing in x and Δt the spacing in t , as shown in Figure 3.2. Note that $h = \Delta t$ throughout. Later k and l are used for y and z in a similar way. When n, j, k, l are used for other purposes (which is sometimes necessary), local context should make the meaning obvious.

The convention for subscript and superscript indexing is as follows:

$$\begin{aligned} u(t + kh) &= u([n + k]h) = u_{n+k} \\ u(x + m\Delta x) &= u([j + m]\Delta x) = u_{j+m} \\ u(x + m\Delta x, t + kh) &= u_{j+m}^{(n+k)} \end{aligned} \quad (3.3)$$

Notice that when used alone, both the time and space indices appear as a subscript, but when used together, time is always a superscript and is usually enclosed with parentheses to distinguish it from an exponent.

Derivatives are expressed according to the usual conventions. Thus for partial derivatives in space or time we use interchangeably

$$\partial_x u = \frac{\partial u}{\partial x}, \quad \partial_t u = \frac{\partial u}{\partial t}, \quad \partial_{xx} u = \frac{\partial^2 u}{\partial x^2}, \quad \text{etc.} \quad (3.4)$$

For the ordinary time derivative in the study of ODE's we use

$$u' = \frac{du}{dt} \quad (3.5)$$

In this text, subscripts on dependent variables are never used to express derivatives. Thus u_x will *not* be used to represent the first derivative of u with respect to x .

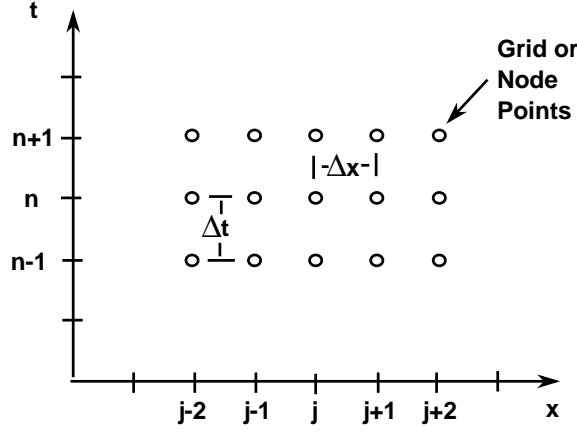


Fig. 3.2. Space-time grid arrangement.

The notation for difference approximations follows the same philosophy, but (with one exception) it is not unique. By this we mean that the symbol δ is used to represent a difference approximation to a derivative such that, for example,

$$\delta_x \approx \partial_x, \quad \delta_{xx} \approx \partial_{xx} \quad (3.6)$$

but the precise nature (and order) of the approximation is not carried in the symbol δ . Other ways are used to determine its precise meaning. The one exception is the symbol Δ , which is defined such that

$$\Delta t_n = t_{n+1} - t_n, \quad \Delta x_j = x_{j+1} - x_j, \quad \Delta u_n = u_{n+1} - u_n, \quad \text{etc.} \quad (3.7)$$

When there is no subscript on Δt or Δx , the spacing is uniform.

3.2 Space Derivative Approximations

A difference approximation can be generated or evaluated by means of a simple Taylor series expansion. For example, consider $u(x, t)$ with t fixed. Then, following the notation convention given in Eqs. 3.1 to 3.3, $x = j\Delta x$ and $u(x + k\Delta x) = u(j\Delta x + k\Delta x) = u_{j+k}$. Expanding the latter term about x gives¹

$$u_{j+k} = u_j + (k\Delta x) \left(\frac{\partial u}{\partial x} \right)_j + \frac{1}{2} (k\Delta x)^2 \left(\frac{\partial^2 u}{\partial x^2} \right)_j + \dots + \frac{1}{n!} (k\Delta x)^n \left(\frac{\partial^n u}{\partial x^n} \right)_j + \dots \quad (3.8)$$

¹ We assume that $u(x, t)$ is continuously differentiable.

Local difference approximations to a given partial derivative can be formed from linear combinations of u_j and u_{j+k} for $k = \pm 1, \pm 2, \dots$

For example, consider the Taylor series expansion for u_{j+1} :

$$u_{j+1} = u_j + (\Delta x) \left(\frac{\partial u}{\partial x} \right)_j + \frac{1}{2} (\Delta x)^2 \left(\frac{\partial^2 u}{\partial x^2} \right)_j + \dots + \frac{1}{n!} (\Delta x)^n \left(\frac{\partial^n u}{\partial x^n} \right)_j + \dots \quad (3.9)$$

Now subtract u_j and divide by Δx to obtain

$$\frac{u_{j+1} - u_j}{\Delta x} = \left(\frac{\partial u}{\partial x} \right)_j + \frac{1}{2} (\Delta x) \left(\frac{\partial^2 u}{\partial x^2} \right)_j + \dots \quad (3.10)$$

Thus the expression $(u_{j+1} - u_j)/\Delta x$ is a reasonable approximation for $\left(\frac{\partial u}{\partial x} \right)_j$ as long as Δx is small relative to some pertinent length scale. Next consider the space difference approximation $(u_{j+1} - u_{j-1})/(2\Delta x)$. Expand the terms in the numerator about j and regroup the result to form the following equation

$$\frac{u_{j+1} - u_{j-1}}{2\Delta x} - \left(\frac{\partial u}{\partial x} \right)_j = \frac{1}{6} \Delta x^2 \left(\frac{\partial^3 u}{\partial x^3} \right)_j + \frac{1}{120} \Delta x^4 \left(\frac{\partial^5 u}{\partial x^5} \right)_j \dots \quad (3.11)$$

When expressed in this manner, it is clear that the discrete terms on the left side of the equation represent a first derivative with a certain amount of error which appears on the right side of the equal sign. It is also clear that the error depends on the grid spacing to a certain order. The error term containing the grid spacing to the lowest power gives the order of the method. From Eq. 3.10, we see that the expression $(u_{j+1} - u_j)/\Delta x$ is a first-order approximation to $\left(\frac{\partial u}{\partial x} \right)_j$. Similarly, Eq. 3.11 shows that $(u_{j+1} - u_{j-1})/(2\Delta x)$ is a second-order approximation to a first derivative. The latter is referred to as the three-point centered difference approximation, and one often sees the summary result presented in the form

$$\left(\frac{\partial u}{\partial x} \right)_j = \frac{u_{j+1} - u_{j-1}}{2\Delta x} + O(\Delta x^2) \quad (3.12)$$

3.3 Finite-Difference Operators

3.3.1 Point Difference Operators

Perhaps the most common examples of finite-difference formulas are the three-point centered-difference approximations for the first and second derivatives:²

² We will derive the second-derivative operator shortly.

$$\left(\frac{\partial u}{\partial x}\right)_j = \frac{1}{2\Delta x}(u_{j+1} - u_{j-1}) + O(\Delta x^2) \quad (3.13)$$

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_j = \frac{1}{\Delta x^2}(u_{j+1} - 2u_j + u_{j-1}) + O(\Delta x^2) \quad (3.14)$$

These are the basis for *point difference operators* since they give an approximation to a derivative at one discrete point in a mesh in terms of surrounding points. However, neither of these expressions tells us how other points in the mesh are differenced or how boundary conditions are enforced. Such additional information requires a more sophisticated formulation.

3.3.2 Matrix Difference Operators

Consider the relation

$$(\delta_{xx}u)_j = \frac{1}{\Delta x^2}(u_{j+1} - 2u_j + u_{j-1}) \quad (3.15)$$

which is a point difference approximation to a second derivative. Now let us derive a *matrix* operator representation for the same approximation. Consider the four-point mesh with boundary points at a and b shown below. Notice that when we speak of “the number of points in a mesh”, we mean the number of *interior* points excluding the boundaries.

	a	1	2	3	4	b
$x =$	0	—	—	—	—	π
$j =$		1	·	·	M	

Four point mesh. $\Delta x = \pi/(M+1)$

Now impose Dirichlet boundary conditions, $u(0) = u_a$, $u(\pi) = u_b$ and use the centered difference approximation given by Eq. 3.15 at every point in the mesh. We arrive at the four equations:

$$\begin{aligned} (\delta_{xx}u)_1 &= \frac{1}{\Delta x^2}(u_a - 2u_1 + u_2) \\ (\delta_{xx}u)_2 &= \frac{1}{\Delta x^2}(u_1 - 2u_2 + u_3) \\ (\delta_{xx}u)_3 &= \frac{1}{\Delta x^2}(u_2 - 2u_3 + u_4) \\ (\delta_{xx}u)_4 &= \frac{1}{\Delta x^2}(u_3 - 2u_4 + u_b) \end{aligned} \quad (3.16)$$

Writing these equations in the more suggestive form

$$\begin{aligned}
(\delta_{xx}u)_1 &= (u_a - 2u_1 + u_2)/\Delta x^2 \\
(\delta_{xx}u)_2 &= (u_1 - 2u_2 + u_3)/\Delta x^2 \\
(\delta_{xx}u)_3 &= (u_2 - 2u_3 + u_4)/\Delta x^2 \\
(\delta_{xx}u)_4 &= (u_3 - 2u_4 + u_b)/\Delta x^2
\end{aligned} \tag{3.17}$$

it is clear that we can express them in a vector-matrix form, and further, that the resulting matrix has a very special form. Introducing

$$\vec{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}, \quad (\vec{bc}) = \frac{1}{\Delta x^2} \begin{bmatrix} u_a \\ 0 \\ 0 \\ u_b \end{bmatrix} \tag{3.18}$$

and

$$A = \frac{1}{\Delta x^2} \begin{bmatrix} -2 & 1 & & \\ 1 & -2 & 1 & \\ & 1 & -2 & 1 \\ & & 1 & -2 \end{bmatrix} \tag{3.19}$$

we can rewrite Eq. 3.17 as

$$\delta_{xx}\vec{u} = A\vec{u} + (\vec{bc}) \tag{3.20}$$

This example illustrates a matrix difference operator. Each line of a matrix difference operator is based on a point difference operator, but the point operators used from line to line are not necessarily the same. For example, boundary conditions may dictate that the lines at or near the bottom or top of the matrix be modified. In the extreme case of the matrix difference operator representing a spectral method, none of the lines is the same. The matrix operators representing the three-point central-difference approximations for a first and second derivative with Dirichlet boundary conditions on a four-point mesh are

$$\delta_x = \frac{1}{2\Delta x} \begin{bmatrix} 0 & 1 & & \\ -1 & 0 & 1 & \\ & -1 & 0 & 1 \\ & & -1 & 0 \end{bmatrix}, \quad \delta_{xx} = \frac{1}{\Delta x^2} \begin{bmatrix} -2 & 1 & & \\ 1 & -2 & 1 & \\ & 1 & -2 & 1 \\ & & 1 & -2 \end{bmatrix} \tag{3.21}$$

As a further example, replace the fourth line in Eq. 3.16 by the following point operator for a Neumann boundary condition (See Section 3.6.):

$$(\delta_{xx}u)_4 = \frac{2}{3} \frac{1}{\Delta x} \left(\frac{\partial u}{\partial x} \right)_b - \frac{2}{3} \frac{1}{\Delta x^2} (u_4 - u_3) \tag{3.22}$$

where the boundary condition is

$$\left(\frac{\partial u}{\partial x} \right)_{x=\pi} = \left(\frac{\partial u}{\partial x} \right)_b \tag{3.23}$$

Then the matrix operator for a three-point central-differencing scheme at interior points and a *second-order approximation* for a Neumann condition on the right is given by

$$\delta_{xx} = \frac{1}{\Delta x^2} \begin{bmatrix} -2 & 1 & & \\ 1 & -2 & 1 & \\ & 1 & -2 & 1 \\ & & 2/3 & -2/3 \end{bmatrix} \quad (3.24)$$

Each of these matrix difference operators is a square matrix with elements that are all zeros except for those along bands which are clustered around the central diagonal. We call such a matrix a *banded matrix* and introduce the notation

$$B(M : a, b, c) = \begin{bmatrix} b & c & & \\ a & b & c & \\ & \ddots & & \\ & & a & b & c \\ & & & a & b \end{bmatrix} \begin{matrix} 1 \\ \\ \\ \vdots \\ M \end{matrix} \quad (3.25)$$

where the matrix dimensions are $M \times M$. Use of M in the argument is optional, and the illustration is given for a simple *tridiagonal* matrix although any number of bands is a possibility. A tridiagonal matrix without constants along the bands can be expressed as $B(\vec{a}, \vec{b}, \vec{c})$. The arguments for a banded matrix are always odd in number, and the central one *always* refers to the central diagonal.

We can now generalize our previous examples. Defining \vec{u} as³

$$\vec{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_M \end{bmatrix} \quad (3.26)$$

we can approximate the second derivative of \vec{u} by

$$\delta_{xx} \vec{u} = \frac{1}{\Delta x^2} B(1, -2, 1) \vec{u} + (\vec{bc}) \quad (3.27)$$

where (\vec{bc}) stands for the vector holding the Dirichlet boundary conditions on the left and right sides:

$$(\vec{bc}) = \frac{1}{\Delta x^2} [u_a, 0, \dots, 0, u_b]^T \quad (3.28)$$

³ Note that \vec{u} is a function of time only, since each element corresponds to one specific spatial location.

If we prescribe Neumann boundary conditions on the right side, as in Eq. 3.24, we find

$$\delta_{xx}\vec{u} = \frac{1}{\Delta x^2} B(\vec{a}, \vec{b}, 1)\vec{u} + (\vec{bc}) \quad (3.29)$$

where

$$\begin{aligned} \vec{a} &= [1, 1, \dots, 2/3]^T \\ \vec{b} &= [-2, -2, -2, \dots, -2/3]^T \\ (\vec{bc}) &= \frac{1}{\Delta x^2} \left[u_a, 0, 0, \dots, \frac{2\Delta x}{3} \left(\frac{\partial u}{\partial x} \right)_b \right]^T \end{aligned}$$

Notice that the matrix operators given by Eqs. 3.27 and 3.29 carry more information than the point operator given by Eq. 3.15. In Eqs. 3.27 and 3.29, the boundary conditions have been uniquely specified and it is clear that the same point operator has been applied at every point in the field except at the boundaries. The ability to specify in the matrix derivative operator the exact nature of the approximation at the various points in the field including the boundaries permits the use of quite general constructions which will be useful later in considerations of stability.

Since we make considerable use of both matrix and point operators, it is important to establish a relation between them. A point operator is generally written for some derivative at the reference point j in terms of neighboring values of the function. For example

$$(\delta_x u)_j = a_2 u_{j-2} + a_1 u_{j-1} + b u_j + c_1 u_{j+1} \quad (3.30)$$

might be the point operator for a first derivative. The corresponding matrix operator has for its arguments the coefficients giving the weights to the values of the function at the various locations. A j -shift in the point operator corresponds to a diagonal shift in the matrix operator. Thus the matrix equivalent of Eq. 3.30 is

$$\delta_x \vec{u} = B(a_2, a_1, b, c_1, 0) \vec{u} \quad (3.31)$$

Note the addition of a zero in the fifth element which makes it clear that b is the coefficient of u_j .

3.3.3 Periodic Matrices

The above illustrated cases in which the boundary conditions are fixed. If the boundary conditions are *periodic*, the form of the matrix operator changes. Consider the eight-point periodic mesh shown below. This can either be presented on a linear mesh with repeated entries, or more suggestively on a circular mesh as in Figure 3.3. When the mesh is laid out on the perimeter of

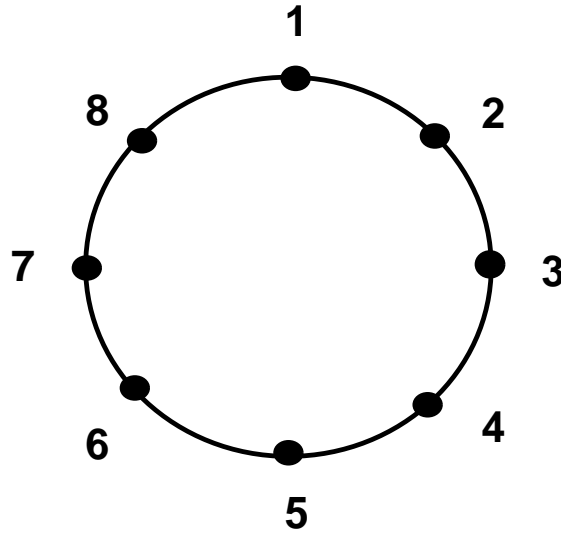


Fig. 3.3. Eight points on a circular mesh.

a circle, it does not matter where the numbering starts, as long as it “ends” at the point just preceding its starting location.

...	7	8	1	2	3	4	5	6	7	8	1	2	...
$x =$	—	—	0	—	—	—	—	—	—	—	2π	—	
$j =$			0	1	M		

Eight points on a linear periodic mesh. $\Delta x = 2\pi/M$

The matrix that represents differencing schemes for scalar equations on a periodic mesh is referred to as a *periodic* matrix. A typical periodic tridiagonal matrix operator with nonuniform entries is given for a 6-point mesh by

$$B_p(6 : \vec{a}, \vec{b}, \vec{c}) = \begin{bmatrix} b_1 & c_2 & & & & a_6 \\ a_1 & b_2 & c_3 & & & \\ & a_2 & b_3 & c_4 & & \\ & & a_3 & b_4 & c_5 & \\ & & & a_4 & b_5 & c_6 \\ c_1 & & & & a_5 & b_6 \end{bmatrix} \quad (3.32)$$

3.3.4 Circulant Matrices

In general, as shown in the example, the elements along the diagonals of a periodic matrix are not constant. However, a special subset of a periodic

matrix is the circulant matrix, formed when the elements along the various bands are constant. Circulant matrices play a vital role in our analysis. We will have much more to say about them later. The most general circulant matrix of order 4 is

$$\begin{bmatrix} b_0 & b_1 & b_2 & b_3 \\ b_3 & b_0 & b_1 & b_2 \\ b_2 & b_3 & b_0 & b_1 \\ b_1 & b_2 & b_3 & b_0 \end{bmatrix} \quad (3.33)$$

Notice that each row of a circulant matrix is shifted (see Figure 3.3) one element to the right of the one above it. The special case of a tridiagonal circulant matrix is given by

$$B_p(M : a, b, c) = \begin{bmatrix} b & c & & a \\ a & b & c & \\ & \ddots & \ddots & \\ & & a & b & c \\ c & & & a & b \end{bmatrix} \begin{matrix} 1 \\ \\ \\ \vdots \\ M \end{matrix} \quad (3.34)$$

When the standard three-point central-differencing approximations for a first and second derivative (see Eq. 3.21) are used with periodic boundary conditions, they take the form

$$(\delta_x)_p = \frac{1}{2\Delta x} \begin{bmatrix} 0 & 1 & -1 \\ -1 & 0 & 1 \\ & -1 & 0 & 1 \\ 1 & & -1 & 0 \end{bmatrix} = \frac{1}{2\Delta x} B_p(-1, 0, 1)$$

and

$$(\delta_{xx})_p = \frac{1}{\Delta x^2} \begin{bmatrix} -2 & 1 & & 1 \\ 1 & -2 & 1 & \\ & 1 & -2 & 1 \\ 1 & & 1 & -2 \end{bmatrix} = \frac{1}{\Delta x^2} B_p(1, -2, 1) \quad (3.35)$$

Clearly, these special cases of periodic operators are also circulant operators. Later on we take advantage of this special property. Notice that there are no boundary condition vectors since this information is all interior to the matrices themselves.

3.4 Constructing Differencing Schemes of Any Order

3.4.1 Taylor Tables

The Taylor series expansion of functions about a fixed point provides a means for constructing finite-difference point operators of any order. A simple and

straightforward way to carry this out is to construct a “Taylor table,” which makes extensive use of the expansion given by Eq. 3.8. As an example, consider Table 3.1, which represents a Taylor table for an approximation of a second derivative using three values of the function centered about the point at which the derivative is to be evaluated.

The table is constructed so that some of the algebra is simplified. At the top of the table we see an expression with a question mark. This represents one of the questions that a study of this table can answer; namely, what is the local error caused by the use of this approximation? Notice that all of the terms in the equation appear in a column at the left of the table (although, in this case, Δx^2 has been multiplied into each term in order to simplify the terms to be put into the table). Then notice that at the head of each column there appears the common factor that occurs in the expansion of each term about the point j , that is,

$$\Delta x^k \cdot \left(\frac{\partial^k u}{\partial x^k} \right)_j, \quad k = 0, 1, 2, \dots$$

The columns to the right of the leftmost one, under the headings, make up the Taylor table. Each entry is the coefficient of the term at the top of the corresponding column in the Taylor series expansion of the term to the left of the corresponding row. For example, the last row in the table corresponds to the Taylor series expansion of $-c u_{j+1}$:

$$\begin{aligned} -c u_{j+1} &= -c u_j - c \cdot (1) \cdot \frac{1}{1} \Delta x \cdot \left(\frac{\partial u}{\partial x} \right)_j - c \cdot (1)^2 \cdot \frac{1}{2} \Delta x^2 \cdot \left(\frac{\partial^2 u}{\partial x^2} \right)_j \\ &\quad - c \cdot (1)^3 \cdot \frac{1}{6} \Delta x^3 \cdot \left(\frac{\partial^3 u}{\partial x^3} \right)_j - c \cdot (1)^4 \cdot \frac{1}{24} \Delta x^4 \cdot \left(\frac{\partial^4 u}{\partial x^4} \right)_j - \dots \end{aligned} \quad (3.36)$$

A Taylor table is simply a convenient way of forming linear combinations of Taylor series on a term by term basis.

Consider the sum of each of these columns. To maximize the order of accuracy of the method, we proceed from left to right and force, by the proper choice of a , b , and c , these sums to be zero. One can easily show that the sums of the first three columns are zero if we satisfy the equation

$$\begin{bmatrix} -1 & -1 & -1 \\ 1 & 0 & -1 \\ -1 & 0 & -1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -2 \end{bmatrix}$$

The solution is given by $[a, b, c] = [1, -2, 1]$.

The columns that do not sum to zero constitute the error.

We designate the first non-vanishing sum to be er_t , and refer to it as the Taylor series error.

	$\left(\frac{\partial^2 u}{\partial x^2}\right)_j - \frac{1}{\Delta x^2}(a u_{j-1} + b u_j + c u_{j+1}) = ?$				
	u_j	$\frac{\Delta x \cdot}{\left(\frac{\partial u}{\partial x}\right)_j}$	$\frac{\Delta x^2 \cdot}{\left(\frac{\partial^2 u}{\partial x^2}\right)_j}$	$\frac{\Delta x^3 \cdot}{\left(\frac{\partial^3 u}{\partial x^3}\right)_j}$	$\frac{\Delta x^4 \cdot}{\left(\frac{\partial^4 u}{\partial x^4}\right)_j}$
$\Delta x^2 \cdot \left(\frac{\partial^2 u}{\partial x^2}\right)_j$			1		
$-a \cdot u_{j-1}$	$-a$	$-a \cdot (-1) \cdot \frac{1}{1}$	$-a \cdot (-1)^2 \cdot \frac{1}{2}$	$-a \cdot (-1)^3 \cdot \frac{1}{6}$	$-a \cdot (-1)^4 \cdot \frac{1}{24}$
$-b \cdot u_j$	$-b$				
$-c \cdot u_{j+1}$	$-c$	$-c \cdot (1) \cdot \frac{1}{1}$	$-c \cdot (1)^2 \cdot \frac{1}{2}$	$-c \cdot (1)^3 \cdot \frac{1}{6}$	$-c \cdot (1)^4 \cdot \frac{1}{24}$
	—	—	—	—	—

Table 3.1. Taylor table for centered 3-point Lagrangian approximation to a second derivative.

In this case er_t occurs at the fifth column in the table (for this example all even columns will vanish by symmetry) and one finds

$$er_t = \frac{1}{\Delta x^2} \left[\frac{-a}{24} + \frac{-c}{24} \right] \Delta x^4 \left(\frac{\partial^4 u}{\partial x^4} \right)_j = \frac{-\Delta x^2}{12} \left(\frac{\partial^4 u}{\partial x^4} \right)_j \quad (3.37)$$

Note that Δx^2 has been divided through to make the error term consistent. We have just derived the familiar 3-point central-differencing point operator for a second derivative

$$\left(\frac{\partial^2 u}{\partial x^2} \right)_j - \frac{1}{\Delta x^2} (u_{j-1} - 2u_j + u_{j+1}) = O(\Delta x^2) \quad (3.38)$$

The Taylor table for a 3-point backward-differencing operator representing a first derivative is shown in Table 3.2. This time the first three columns sum to zero if

$$\begin{bmatrix} -1 & -1 & -1 \\ 2 & 1 & 0 \\ -4 & -1 & 0 \end{bmatrix} \begin{bmatrix} a_2 \\ a_1 \\ b \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}$$

which gives $[a_2, a_1, b] = \frac{1}{2}[1, -4, 3]$. In this case the fourth column provides the leading truncation error term:

$$er_t = \frac{1}{\Delta x} \left[\frac{8a_2}{6} + \frac{a_1}{6} \right] \Delta x^3 \left(\frac{\partial^3 u}{\partial x^3} \right)_j = \frac{\Delta x^2}{3} \left(\frac{\partial^3 u}{\partial x^3} \right)_j \quad (3.39)$$

Thus we have derived a second-order backward-difference approximation of a first derivative:

$$\left(\frac{\partial u}{\partial x} \right)_j - \frac{1}{2\Delta x} (u_{j-2} - 4u_{j-1} + 3u_j) = O(\Delta x^2) \quad (3.40)$$

3.4.2 Generalization of Difference Formulas

In general, a difference approximation to the m th derivative at grid point j can be cast in terms of $q + p + 1$ neighboring points as

$$\left(\frac{\partial^m u}{\partial x^m} \right)_j - \sum_{i=-p}^q a_i u_{j+i} = er_t \quad (3.41)$$

where the a_i are coefficients to be determined through the use of Taylor tables to produce approximations of a given order. Clearly this process can be used to find forward, backward, skewed, or central point operators of any order for any derivative. It could be computer automated and extended to higher dimensions. More important, however, is the fact that it can be further generalized. In order to do this, let us approach the subject in a slightly different way, that is from the point of view of interpolation formulas. These formulas are discussed in many texts on numerical analysis.

$$\left(\frac{\partial u}{\partial x}\right)_j - \frac{1}{\Delta x}(a_2 u_{j-2} + a_1 u_{j-1} + b u_j) = ?$$

	u_j	$\frac{\Delta x \cdot}{\left(\frac{\partial u}{\partial x}\right)_j}$	$\frac{\Delta x^2 \cdot}{\left(\frac{\partial^2 u}{\partial x^2}\right)_j}$	$\frac{\Delta x^3 \cdot}{\left(\frac{\partial^3 u}{\partial x^3}\right)_j}$	$\frac{\Delta x^4 \cdot}{\left(\frac{\partial^4 u}{\partial x^4}\right)_j}$
$\Delta x \cdot \left(\frac{\partial u}{\partial x}\right)_j$		1			
$-a_2 \cdot u_{j-2}$	$-a_2$	$-a_2 \cdot (-2) \cdot \frac{1}{1}$	$-a_2 \cdot (-2)^2 \cdot \frac{1}{2}$	$-a_2 \cdot (-2)^3 \cdot \frac{1}{6}$	$-a_2 \cdot (-2)^4 \cdot \frac{1}{24}$
$-a_1 \cdot u_{j-1}$	$-a_1$	$-a_1 \cdot (-1) \cdot \frac{1}{1}$	$-a_1 \cdot (-1)^2 \cdot \frac{1}{2}$	$-a_1 \cdot (-1)^3 \cdot \frac{1}{6}$	$-a_1 \cdot (-1)^4 \cdot \frac{1}{24}$
$-b \cdot u_j$	$-b$				
	—	—————	—————	—————	—————

Table 3.2. Taylor table for backward 3-point Lagrangian approximation to a first derivative.

3.4.3 Lagrange and Hermite Interpolation Polynomials

The Lagrangian interpolation polynomial is given by

$$u(x) = \sum_{k=0}^K a_k(x) u_k \quad (3.42)$$

where $a_k(x)$ are polynomials in x of degree K . The construction of the $a_k(x)$ can be taken from the simple Lagrangian formula for quadratic interpolation (or extrapolation) with non-equispaced points

$$\begin{aligned} u(x) = u_0 \frac{(x_1 - x)(x_2 - x)}{(x_1 - x_0)(x_2 - x_0)} &+ u_1 \frac{(x_0 - x)(x_2 - x)}{(x_0 - x_1)(x_2 - x_1)} \\ &+ u_2 \frac{(x_0 - x)(x_1 - x)}{(x_0 - x_2)(x_1 - x_2)} \end{aligned} \quad (3.43)$$

Notice that the coefficient of each u_k is one when $x = x_k$, and zero when x takes any other discrete value in the set. If we take the first or second derivative of $u(x)$, impose an equispaced mesh, and evaluate these derivatives at the appropriate discrete point, we rederive the finite-difference approximations just presented. Finite-difference schemes that can be derived from Eq. 3.42 are referred to as Lagrangian approximations.

A generalization of the Lagrangian approach is brought about by using Hermitian interpolation. To construct a polynomial for $u(x)$, Hermite formulas use values of the function *and its derivative(s)* at given points in space. Our illustration is for the case in which discrete values of the function and its first derivative are used, producing the expression

$$u(x) = \sum a_k(x) u_k + \sum b_k(x) \left(\frac{\partial u}{\partial x} \right)_k \quad (3.44)$$

Obviously higher-order derivatives could be included as the problems dictate. A complete discussion of these polynomials can be found in many references on numerical methods, but here we need only the concept.

The previous examples of a Taylor table constructed *explicit* point difference operators from Lagrangian interpolation formulas. Consider next the Taylor table for an *implicit* space differencing scheme for a first derivative arising from the use of an Hermite interpolation formula. A generalization of Eq. 3.41 can include derivatives at neighboring points, i.e.,

$$\sum_{i=-r}^s b_i \left(\frac{\partial^m u}{\partial x^m} \right)_{j+i} - \sum_{i=-p}^q a_i u_{j+i} = er_t \quad (3.45)$$

analogous to Eq. 3.44. An example formula is illustrated at the top of Table 3.3. Here not only is the derivative at point j represented, but also included

are derivatives at points $j - 1$ and $j + 1$, which also must be expanded using Taylor series about point j . This requires the following generalization of the Taylor series expansion given in Eq. 3.8:

$$\left(\frac{\partial^m u}{\partial x^m}\right)_{j+k} = \left\{ \left[\sum_{n=0}^{\infty} \frac{1}{n!} (k\Delta x)^n \frac{\partial^n}{\partial x^n} \right] \left(\frac{\partial^m u}{\partial x^m}\right) \right\}_j \quad (3.46)$$

The derivative terms now have coefficients (the coefficient on the j point is taken as one to simplify the algebra) which must be determined using the Taylor table approach as outlined below.

To maximize the order of accuracy, we must satisfy the relation

$$\begin{bmatrix} -1 & -1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 1 & 1 \\ -1 & 0 & -1 & -2 & 2 \\ 1 & 0 & -1 & 3 & 3 \\ -1 & 0 & -1 & -4 & 4 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

having the solution $[a, b, c, d, e] = \frac{1}{4}[-3, 0, 3, 1, 1]$. Under these conditions, the sixth column sums to

$$er_t = \frac{\Delta x^4}{120} \left(\frac{\partial^5 u}{\partial x^5} \right)_j \quad (3.47)$$

and the method can be expressed as

$$\left(\frac{\partial u}{\partial x}\right)_{j-1} + 4\left(\frac{\partial u}{\partial x}\right)_j + \left(\frac{\partial u}{\partial x}\right)_{j+1} - \frac{3}{\Delta x}(-u_{j-1} + u_{j+1}) = O(\Delta x^4) \quad (3.48)$$

This is also referred to as a *Padé* or *compact* formula.

3.4.4 Practical Application of Padé Formulas

It is one thing to construct methods using the Hermitian concept and quite another to implement them in a computer code. In the form of a point operator it is probably not evident at first just how Eq. 3.48 can be applied. However, the situation is quite easy to comprehend if we express the same method in the form of a matrix operator. A banded matrix notation for Eq. 3.48 is

$$\frac{1}{6}B(1, 4, 1)\delta_x \vec{u} = \frac{1}{2\Delta x}B(-1, 0, 1)\vec{u} + \begin{pmatrix} \vec{bc} \end{pmatrix} \quad (3.49)$$

in which Dirichlet boundary conditions have been imposed.⁴ Mathematically this is equivalent to

⁴ In this case the vector containing the boundary conditions would include values of both u and $\partial u / \partial x$ at both boundaries.

$$d\left(\frac{\partial u}{\partial x}\right)_{j-1} + \left(\frac{\partial u}{\partial x}\right)_j + e\left(\frac{\partial u}{\partial x}\right)_{j+1} - \frac{1}{\Delta x}(au_{j-1} + bu_j + cu_{j+1}) = ?$$

	u_j	$\frac{\Delta x \cdot}{\left(\frac{\partial u}{\partial x}\right)_j}$	$\frac{\Delta x^2 \cdot}{\left(\frac{\partial^2 u}{\partial x^2}\right)_j}$	$\frac{\Delta x^3 \cdot}{\left(\frac{\partial^3 u}{\partial x^3}\right)_j}$	$\frac{\Delta x^4 \cdot}{\left(\frac{\partial^4 u}{\partial x^4}\right)_j}$	$\frac{\Delta x^5 \cdot}{\left(\frac{\partial^5 u}{\partial x^5}\right)_j}$
$\Delta x \cdot d\left(\frac{\partial u}{\partial x}\right)_{j-1}$		d	$d \cdot (-1) \cdot \frac{1}{1}$	$d \cdot (-1)^2 \cdot \frac{1}{2}$	$d \cdot (-1)^3 \cdot \frac{1}{6}$	$d \cdot (-1)^4 \cdot \frac{1}{24}$
$\Delta x \cdot \left(\frac{\partial u}{\partial x}\right)_j$		1				
$\Delta x \cdot e\left(\frac{\partial u}{\partial x}\right)_{j+1}$		e	$e \cdot (1) \cdot \frac{1}{1}$	$e \cdot (1)^2 \cdot \frac{1}{2}$	$e \cdot (1)^3 \cdot \frac{1}{6}$	$e \cdot (1)^4 \cdot \frac{1}{24}$
$-a \cdot u_{j-1}$	$-a$	$-a \cdot (-1) \cdot \frac{1}{1}$	$-a \cdot (-1)^2 \cdot \frac{1}{2}$	$-a \cdot (-1)^3 \cdot \frac{1}{6}$	$-a \cdot (-1)^4 \cdot \frac{1}{24}$	$-a \cdot (-1)^5 \cdot \frac{1}{120}$
$-b \cdot u_j$	$-b$					
$-c \cdot u_{j+1}$	$-c$	$-c \cdot (1) \cdot \frac{1}{1}$	$-c \cdot (1)^2 \cdot \frac{1}{2}$	$-c \cdot (1)^3 \cdot \frac{1}{6}$	$-c \cdot (1)^4 \cdot \frac{1}{24}$	$-c \cdot (1)^5 \cdot \frac{1}{120}$
	<hr/>	<hr/>	<hr/>	<hr/>	<hr/>	<hr/>

Table 3.3. Taylor table for central 3-point Hermitian approximation to a first derivative.

$$\delta_x \vec{u} = 6[B(1, 4, 1)]^{-1} \left[\frac{1}{2\Delta x} B(-1, 0, 1) \vec{u} + \begin{pmatrix} \vec{bc} \end{pmatrix} \right] \quad (3.50)$$

which can be reexpressed by the “predictor-corrector” sequence

$$\begin{aligned} \vec{u} &= \frac{1}{2\Delta x} B(-1, 0, 1) \vec{u} + \begin{pmatrix} \vec{bc} \end{pmatrix} \\ \delta_x \vec{u} &= 6[B(1, 4, 1)]^{-1} \vec{u} \end{aligned} \quad (3.51)$$

With respect to practical implementation, the meaning of the predictor in this sequence should be clear. It simply says – take the vector array \vec{u} , difference it, add the boundary conditions, and store the result in the intermediate array \vec{u} . The meaning of the second row is more subtle, since it is demanding the evaluation of an inverse operator, but it still can be given a simple interpretation. An inverse matrix operator implies the solution of a coupled set of linear equations. These operators are very common in finite difference applications. They appear in the form of banded matrices having a small bandwidth, in this case a tridiagonal. The evaluation of $[B(1, 4, 1)]^{-1}$ is found by means of a tridiagonal “solver”, which is simple to code, efficient to run, and widely used. In general, Hermitian or Padé approximations can be practical when they can be implemented by the use of efficient banded solvers.

3.4.5 Other Higher-Order Schemes

Hermitian forms of the second derivative can also be easily derived by means of a Taylor table. For example

$$\delta_{xx} \vec{u} = 12[B(1, 10, 1)]^{-1} \left[\frac{1}{\Delta x^2} B(1, -2, 1) \vec{u} + \begin{pmatrix} \vec{bc} \end{pmatrix} \right] \quad (3.52)$$

is $O(\Delta x^4)$ and makes use of only tridiagonal operations. It should be mentioned that the spline approximation is one form of a Padé matrix difference operator. It is given by

$$\delta_{xx} \vec{u} = 6[B(1, 4, 1)]^{-1} \left[\frac{1}{\Delta x^2} B(1, -2, 1) \vec{u} + \begin{pmatrix} \vec{bc} \end{pmatrix} \right] \quad (3.53)$$

but its order of accuracy is only $O(\Delta x^2)$. How much this reduction in accuracy is offset by the increased global continuity built into a spline fit is not known. We note that the spline fit of a first derivative is identical to any of the expressions in Eqs. 3.48 to 3.51.

A final word on Hermitian approximations. Clearly they have an advantage over 3-point Lagrangian schemes because of their increased accuracy.

However, a more subtle point is that they get this increase in accuracy using information that is still local to the point where the derivatives are being evaluated. In application, this can be advantageous at boundaries and in the vicinity of steep gradients. It is obvious, of course, that five-point schemes using Lagrangian approximations can be derived that have the same order of accuracy as the methods given in Eqs. 3.48 and 3.52, but they will have a wider spread of space indices. In particular, two Lagrangian schemes with the same order of accuracy are (here we ignore the problem created by the boundary conditions, although this is one of the principal issues in applying these schemes):

$$\frac{\partial \vec{u}}{\partial x} - \frac{1}{12\Delta x} B_p(1, -8, 0, 8, -1) \vec{u} = O(\Delta x^4) \quad (3.54)$$

$$\frac{\partial^2 \vec{u}}{\partial x^2} - \frac{1}{12\Delta x^2} B_p(-1, 16, -30, 16, -1) \vec{u} = O(\Delta x^4) \quad (3.55)$$

3.5 Fourier Error Analysis

In order to select a finite-difference scheme for a given application one must be able to assess the accuracy of the candidate schemes. The accuracy of an operator is often expressed in terms of the order of the leading error term determined from a Taylor table. While this is a useful measure, it provides a fairly limited description. Further information about the error behavior of a finite-difference scheme can be obtained using Fourier error analysis.

3.5.1 Application to a Spatial Operator

An arbitrary periodic function can be decomposed into its Fourier components, which are in the form $e^{i\kappa x}$, where κ is the wavenumber. It is therefore of interest to examine how well a given finite-difference operator approximates derivatives of $e^{i\kappa x}$. We will concentrate here on first derivative approximations, although the analysis is equally applicable to higher derivatives.

The exact first derivative of $e^{i\kappa x}$ is

$$\frac{\partial e^{i\kappa x}}{\partial x} = i\kappa e^{i\kappa x} \quad (3.56)$$

If we apply, for example, a second-order centered difference operator to $u_j = e^{i\kappa x_j}$, where $x_j = j\Delta x$, we get

$$\begin{aligned} (\delta_x u)_j &= \frac{u_{j+1} - u_{j-1}}{2\Delta x} \\ &= \frac{e^{i\kappa\Delta x(j+1)} - e^{i\kappa\Delta x(j-1)}}{2\Delta x} \end{aligned}$$

$$\begin{aligned}
&= \frac{(e^{i\kappa\Delta x} - e^{-i\kappa\Delta x})e^{i\kappa x_j}}{2\Delta x} \\
&= \frac{1}{2\Delta x}[(\cos \kappa\Delta x + i \sin \kappa\Delta x) - (\cos \kappa\Delta x - i \sin \kappa\Delta x)]e^{i\kappa x_j} \\
&= i \frac{\sin \kappa\Delta x}{\Delta x} e^{i\kappa x_j} \\
&= i\kappa^* e^{i\kappa x_j}
\end{aligned} \tag{3.57}$$

where κ^* is the modified wavenumber. The modified wavenumber is so named because it appears where the wavenumber, κ , appears in the exact expression. Thus the degree to which the modified wavenumber approximates the actual wavenumber is a measure of the accuracy of the approximation.

For the second-order centered difference operator the modified wavenumber is given by

$$\kappa^* = \frac{\sin \kappa\Delta x}{\Delta x} \tag{3.58}$$

Note that κ^* approximates κ to second-order accuracy, as is to be expected, since

$$\frac{\sin \kappa\Delta x}{\Delta x} = \kappa - \frac{\kappa^3 \Delta x^2}{6} + \dots$$

Equation 3.58 is plotted in Figure 3.4, along with similar relations for the standard fourth-order centered difference scheme and the fourth-order Padé scheme. The expression for the modified wavenumber provides the accuracy with which a given wavenumber component of the solution is resolved for the entire wavenumber range available in a mesh of a given size, $0 \leq \kappa\Delta x \leq \pi$.

In general, finite-difference operators can be written in the form

$$(\delta_x)_j = (\delta_x^a)_j + (\delta_x^s)_j$$

where $(\delta_x^a)_j$ is an antisymmetric operator and $(\delta_x^s)_j$ is a symmetric operator.⁵ If we restrict our interest to schemes extending from $j - 3$ to $j + 3$, then

$$(\delta_x^a u)_j = \frac{1}{\Delta x}[a_1(u_{j+1} - u_{j-1}) + a_2(u_{j+2} - u_{j-2}) + a_3(u_{j+3} - u_{j-3})]$$

and

$$(\delta_x^s u)_j = \frac{1}{\Delta x}[d_0 u_j + d_1(u_{j+1} + u_{j-1}) + d_2(u_{j+2} + u_{j-2}) + d_3(u_{j+3} + u_{j-3})]$$

The corresponding modified wavenumber is

$$\begin{aligned}
i\kappa^* &= \frac{1}{\Delta x}[d_0 + 2(d_1 \cos \kappa\Delta x + d_2 \cos 2\kappa\Delta x + d_3 \cos 3\kappa\Delta x) \\
&\quad + 2i(a_1 \sin \kappa\Delta x + a_2 \sin 2\kappa\Delta x + a_3 \sin 3\kappa\Delta x)]
\end{aligned} \tag{3.59}$$

⁵ In terms of a circulant matrix operator A , the antisymmetric part is obtained from $(A - A^T)/2$ and the symmetric part from $(A + A^T)/2$.

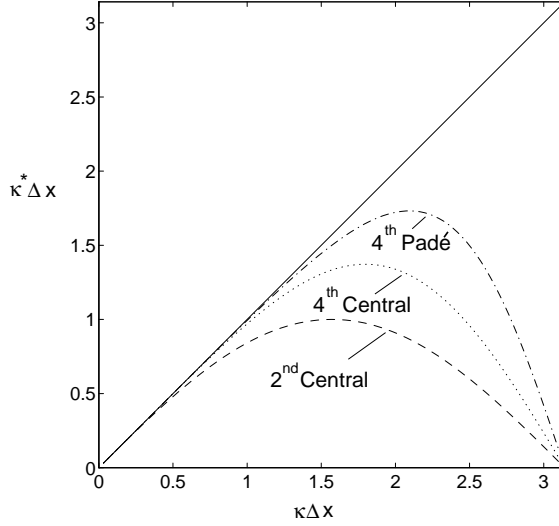


Fig. 3.4. Modified wavenumber for various schemes.

When the finite-difference operator is antisymmetric (centered), the modified wavenumber is purely real. When the operator includes a symmetric component, the modified wavenumber is complex, with the imaginary component being entirely error. The fourth-order Padé scheme is given by

$$(\delta_x u)_{j-1} + 4(\delta_x u)_j + (\delta_x u)_{j+1} = \frac{3}{\Delta x}(u_{j+1} - u_{j-1})$$

The modified wavenumber for this scheme satisfies⁶

$$i\kappa^* e^{-i\kappa\Delta x} + 4i\kappa^* + i\kappa^* e^{i\kappa\Delta x} = \frac{3}{\Delta x}(e^{i\kappa\Delta x} - e^{-i\kappa\Delta x})$$

which gives

$$i\kappa^* = \frac{3i \sin \kappa\Delta x}{(2 + \cos \kappa\Delta x)\Delta x}$$

The modified wavenumber provides a useful tool for assessing difference approximations. In the context of the linear convection equation, the errors can be given a physical interpretation. Consider once again the linear convection equation in the form

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0$$

⁶ Note that terms such as $(\delta_x u)_{j-1}$ are handled by letting $(\delta_x u)_j = i\kappa^* e^{i\kappa j\Delta x}$ and evaluating the shift in j .

on a domain extending from $-\infty$ to ∞ . Recall from Section 2.3.2 that a solution initiated by a harmonic function with wavenumber κ is

$$u(x, t) = f(t)e^{i\kappa x} \quad (3.60)$$

where $f(t)$ satisfies the ODE

$$\frac{df}{dt} = -ia\kappa f$$

Solving for $f(t)$ and substituting into Eq. 3.60 gives the exact solution as

$$u(x, t) = f(0)e^{i\kappa(x-at)}$$

If second-order centered differences are applied to the spatial term, the following ODE is obtained for $f(t)$:

$$\frac{df}{dt} = -ia \left[\frac{\sin \kappa \Delta x}{\Delta x} \right] f = -ia\kappa^* f \quad (3.61)$$

Solving this ODE exactly (since we are considering the error from the spatial approximation only) and substituting into Eq. 3.60, we obtain

$$u_{\text{numerical}}(x, t) = f(0)e^{i\kappa(x-a^*t)} \quad (3.62)$$

where a^* is the numerical (or modified) phase speed, which is related to the modified wavenumber by

$$\frac{a^*}{a} = \frac{\kappa^*}{\kappa}$$

For the above example,

$$\frac{a^*}{a} = \frac{\sin \kappa \Delta x}{\kappa \Delta x}$$

The numerical phase speed is the speed at which a harmonic function is propagated numerically. Since $a^*/a \leq 1$ for this example, the numerical solution propagates too slowly. Since a^* is a function of the wavenumber, the numerical approximation introduces dispersion, although the original PDE is nondispersive. As a result, a waveform consisting of many different wavenumber components eventually loses its original form.

Figure 3.5 shows the numerical phase speed for the schemes considered previously. The number of *points per wavelength (PPW)* by which a given wave is resolved is given by $2\pi/\kappa\Delta x$. The resolving efficiency of a scheme can be expressed in terms of the *PPW* required to produce errors below a specified level. For example, the second-order centered difference scheme requires 80 *PPW* to produce an error in phase speed of less than 0.1 percent. The 5-point fourth-order centered scheme and the fourth-order Padé scheme require 15 and 10 *PPW*, respectively, to achieve the same error level.

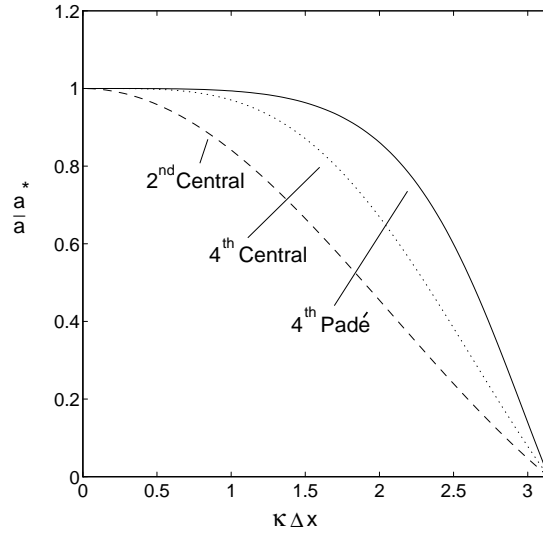


Fig. 3.5. Numerical phase speed for various schemes.

For our example using second-order centered differences, the modified wavenumber is purely real, but in the general case it can include an imaginary component as well, as shown in Eq. 3.59. In that case, the error in the phase speed is determined from the real part of the modified wavenumber, while the imaginary part leads to an error in the amplitude of the solution, as can be seen by inspecting Eq. 3.61. Thus the antisymmetric portion of the spatial difference operator determines the error in speed and the symmetric portion the error in amplitude. This will be further discussed in Section ??.

3.6 Difference Operators at Boundaries

As discussed in Section 3.3.2, a matrix difference operator incorporates both the difference approximation in the interior of the domain and that at the boundaries. In this section, we consider the boundary operators needed for our model equations for convection and diffusion. In the case of periodic boundary conditions, no special boundary operators are required.

3.6.1 The Linear Convection Equation

Referring to Section 2.3, the boundary conditions for the linear convection equation can be either periodic or of inflow-outflow type. In the latter case, a Dirichlet boundary condition is given at the inflow boundary, while no

condition is specified at the outflow boundary. Here we assume that the wave speed a is positive; thus the left-hand boundary is the inflow boundary, and the right-hand boundary is the outflow boundary. The vector of unknowns is

$$\vec{u} = [u_1, u_2, \dots, u_M]^T \quad (3.63)$$

and u_0 is specified.

Consider first the inflow boundary. It is clear that as long as the interior difference approximation does not extend beyond u_{j-1} , then no special treatment is required for this boundary. For example, with second-order centered differences we obtain

$$\delta_x \vec{u} = A\vec{u} + (\vec{bc}) \quad (3.64)$$

with

$$A = \frac{1}{2\Delta x} \begin{bmatrix} 0 & 1 & & \\ -1 & 0 & 1 & \\ & -1 & 0 & 1 \\ & & \ddots & \ddots \end{bmatrix}, \quad (\vec{bc}) = \frac{1}{2\Delta x} \begin{bmatrix} -u_0 \\ 0 \\ 0 \\ \vdots \end{bmatrix} \quad (3.65)$$

However, if we use the fourth-order interior operator given in Eq. 3.54, then the approximation at $j = 1$ requires a value of u_{j-2} , which is outside the domain. Hence, a different operator is required at $j = 1$ which extends only to $j - 1$, while having the appropriate order of accuracy. Such an operator, known as a *numerical boundary scheme*, can have an order of accuracy which is *one order lower* than that of the interior scheme, and the global accuracy will equal that of the interior scheme.⁷ For example, with fourth-order centered differences, we can use the following third-order operator at $j = 1$:

$$(\delta_x u)_1 = \frac{1}{6\Delta x} (-2u_0 - 3u_1 + 6u_2 - u_3) \quad (3.66)$$

which is easily derived using a Taylor table. The resulting difference operator has the form of Eq. 3.64 with

$$A = \frac{1}{12\Delta x} \begin{bmatrix} -6 & 12 & -2 & & \\ -8 & 0 & 8 & -1 & \\ & 1 & -8 & 0 & 8 & -1 \\ & & & \ddots & \ddots & \ddots \end{bmatrix}, \quad (\vec{bc}) = \frac{1}{12\Delta x} \begin{bmatrix} -4u_0 \\ u_0 \\ 0 \\ \vdots \end{bmatrix} \quad (3.67)$$

This approximation is globally fourth-order accurate.

At the outflow boundary, no boundary condition is specified. We must approximate $\partial u / \partial x$ at node M with no information about u_{M+1} . Thus the

⁷ Proof of this theorem is beyond the scope of this book; the interested reader should consult the literature for further details.

second-order centered-difference operator, which requires u_{j+1} , cannot be used at $j = M$. A backward-difference formula must be used. With a second-order interior operator, the following first-order backward formula can be used:

$$(\delta_x u)_M = \frac{1}{\Delta x}(u_M - u_{M-1}) \quad (3.68)$$

This produces a difference operator with

$$A = \frac{1}{2\Delta x} \begin{bmatrix} 0 & 1 & & & \\ -1 & 0 & 1 & & \\ & -1 & 0 & 1 & \\ & & \ddots & \ddots & \\ & & & -1 & 0 & 1 \\ & & & & -2 & 2 \end{bmatrix}, \quad (\vec{bc}) = \frac{1}{2\Delta x} \begin{bmatrix} -u_0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (3.69)$$

In the case of a fourth-order centered interior operator the last two rows of A require modification.

Another approach to the development of boundary schemes is in terms of space extrapolation. The following formula allows u_{M+1} to be extrapolated from the interior data to arbitrary order on an equispaced grid:

$$(1 - E^{-1})^p u_{M+1} = 0 \quad (3.70)$$

where E is the shift operator defined by $Eu_j = u_{j+1}$, and the order of the approximation is $p - 1$. For example, with $p = 2$ we obtain

$$(1 - 2E^{-1} + E^{-2})u_{M+1} = u_{M+1} - 2u_M + u_{M-1} = 0 \quad (3.71)$$

which gives the following first-order approximation to u_{M+1} :

$$u_{M+1} = 2u_M - u_{M-1} \quad (3.72)$$

Substituting this into the second-order centered-difference operator applied at node M gives

$$\begin{aligned} (\delta_x u)_M &= \frac{1}{2\Delta x}(u_{M+1} - u_{M-1}) = \frac{1}{2\Delta x}(2u_M - u_{M-1} - u_{M-1}) \\ &= \frac{1}{\Delta x}(u_M - u_{M-1}) \end{aligned} \quad (3.73)$$

which is identical to Eq. 3.68.

3.6.2 The Diffusion Equation

In solving the diffusion equation, we must consider Dirichlet and Neumann boundary conditions. The treatment of a Dirichlet boundary condition proceeds along the same lines as the inflow boundary for the convection equation discussed above. With the second-order centered interior operator

$$(\delta_{xx}u)_j = \frac{1}{\Delta x^2}(u_{j+1} - 2u_j + u_{j-1}) \quad (3.74)$$

no modifications are required near boundaries, leading to the matrix difference operator given in Eq. 3.27.

For a Neumann boundary condition, we assume that $\partial u / \partial x$ is specified at $j = M + 1$, that is

$$\left(\frac{\partial u}{\partial x}\right)_{M+1} = \left(\frac{\partial u}{\partial x}\right)_b \quad (3.75)$$

Thus we design an operator at node M which is in the following form:

$$(\delta_{xx}u)_M = \frac{1}{\Delta x^2}(au_{M-1} + bu_M) + \frac{c}{\Delta x} \left(\frac{\partial u}{\partial x}\right)_{M+1} \quad (3.76)$$

where a , b , and c are constants which can easily be determined using a Taylor table, as shown in Table 3.4.

Solving for a , b , and c , we obtain the following first-order operator:

$$(\delta_{xx}u)_M = \frac{1}{3\Delta x^2}(2u_{M-1} - 2u_M) + \frac{2}{3\Delta x} \left(\frac{\partial u}{\partial x}\right)_{M+1} \quad (3.77)$$

which produces the matrix difference operator given in Eq. 3.29.

We can also obtain the operator in Eq. 3.77 using the space extrapolation idea. Consider a second-order backward-difference approximation applied at node $M + 1$:

$$\left(\frac{\partial u}{\partial x}\right)_{M+1} = \frac{1}{2\Delta x}(u_{M-1} - 4u_M + 3u_{M+1}) + O(\Delta x^2) \quad (3.78)$$

Solving for u_{M+1} gives

$$u_{M+1} = \frac{1}{3} \left[4u_M - u_{M-1} + 2\Delta x \left(\frac{\partial u}{\partial x}\right)_{M+1} \right] + O(\Delta x^3) \quad (3.79)$$

Substituting this into the second-order centered difference operator for a second derivative applied at node M gives

$$\begin{aligned} (\delta_{xx}u)_M &= \frac{1}{\Delta x^2}(u_{M+1} - 2u_M + u_{M-1}) \\ &= \frac{1}{3\Delta x^2} \left[3u_{M-1} - 6u_M + 4u_M - u_{M-1} + 2\Delta x \left(\frac{\partial u}{\partial x}\right)_{M+1} \right] \\ &= \frac{1}{3\Delta x^2}(2u_{M-1} - 2u_M) + \frac{2}{3\Delta x} \left(\frac{\partial u}{\partial x}\right)_{M+1} \end{aligned} \quad (3.80)$$

which is identical to Eq. 3.77.

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_j - \left[\frac{1}{\Delta x^2}(a^{u_{j-1}} + bu_j) + \frac{c}{\Delta x} \left(\frac{\partial u}{\partial x}\right)_{j+1} \right] = ?$$

	u_j	$\frac{\Delta x \cdot}{\left(\frac{\partial u}{\partial x}\right)_j}$	$\frac{\Delta x^2 \cdot}{\left(\frac{\partial^2 u}{\partial x^2}\right)_j}$	$\frac{\Delta x^3 \cdot}{\left(\frac{\partial^3 u}{\partial x^3}\right)_j}$	$\frac{\Delta x^4 \cdot}{\left(\frac{\partial^4 u}{\partial x^4}\right)_j}$
$\Delta x^2 \cdot \left(\frac{\partial^2 u}{\partial x^2}\right)_j$			1		
$-a \cdot u_{j-1}$	$-a$	$-a \cdot (-1) \cdot \frac{1}{1}$	$-a \cdot (-1)^2 \cdot \frac{1}{2}$	$-a \cdot (-1)^3 \cdot \frac{1}{6}$	$-a \cdot (-1)^4 \cdot \frac{1}{24}$
$-b \cdot u_j$	$-b$				
$-\Delta x \cdot c \cdot \left(\frac{\partial u}{\partial x}\right)_{j+1}$		$-c$	$-c \cdot (1) \cdot \frac{1}{1}$	$-c \cdot (1)^2 \cdot \frac{1}{2}$	$-c \cdot (1)^3 \cdot \frac{1}{6}$
	—	—	—	—	—

Table 3.4. Taylor table for Neumann boundary condition for the diffusion equation.

Exercises

3.1 Derive a third-order finite-difference approximation to a first derivative in the form

$$(\delta_x u)_j = \frac{1}{\Delta x}(au_{j-2} + bu_{j-1} + cu_j + du_{j+1})$$

Find the leading error term.

3.2 Derive a finite-difference approximation to a first derivative in the form

$$a(\delta_x u)_{j-1} + (\delta_x u)_j = \frac{1}{\Delta x}(bu_{j-1} + cu_j + du_{j+1})$$

Find the leading error term.

3.3 Using a 4 (interior) point mesh, write out the 4×4 matrices and the boundary-condition vector formed by using the scheme derived in question 3.2 when both u and $\partial u / \partial x$ are given at $j = 0$ and u is given at $j = 5$.

3.4 Repeat question 3.2 with $d = 0$.

3.5 Derive a finite-difference approximation to a third derivative in the form

$$(\delta_{xxx} u)_j = \frac{1}{\Delta x^3}(au_{j-2} + bu_{j-1} + cu_j + du_{j+1} + eu_{j+2})$$

Find the leading error term.

3.6 Derive a compact (or Padé) finite-difference approximation to a second derivative in the form

$$d(\delta_{xx} u)_{j-1} + (\delta_{xx} u)_j + e(\delta_{xx} u)_{j+1} = \frac{1}{\Delta x^2}(au_{j-1} + bu_j + cu_{j+1})$$

Find the leading error term.

3.7 Find the modified wavenumber for the operator derived in question 3.1. Plot the real and imaginary parts of $\kappa^* \Delta x$ vs. $\kappa \Delta x$ for $0 \leq \kappa \Delta x \leq \pi$. Compare the real part with that obtained from the fourth-order centered operator (Eq. 3.54).

3.8 Application of the second-derivative operator to the function $e^{i\kappa x}$ gives

$$\frac{\partial^2 e^{i\kappa x}}{\partial x^2} = -\kappa^2 e^{i\kappa x}$$

Application of a difference operator for the second derivative gives

$$(\delta_{xx} e^{i\kappa j \Delta x})_j = -\kappa^{*2} e^{i\kappa x}$$

thus defining the modified wavenumber κ^* for a second derivative approximation. Find the modified wavenumber for the second-order centered difference operator for a second derivative, the noncompact fourth-order operator (Eq. 3.55), and the compact fourth-order operator derived in question 3.6. Plot $(\kappa^* \Delta x)^2$ vs. $\kappa \Delta x$ for $0 \leq \kappa \Delta x \leq \pi$.

3.9 Find the grid-points-per-wavelength (PPW) requirement to achieve a phase speed error less than 0.1 percent for sixth-order noncompact and compact centered approximations to a first derivative.

3.10 Consider the following one-sided differencing schemes, which are first-, second-, and third-order, respectively:

$$(\delta_x u)_j = (u_j - u_{j-1}) / \Delta x$$

$$(\delta_x u)_j = (3u_j - 4u_{j-1} + u_{j-2}) / (2\Delta x)$$

$$(\delta_x u)_j = (11u_j - 18u_{j-1} + 9u_{j-2} - 2u_{j-3}) / (6\Delta x)$$

Find the modified wavenumber for each of these schemes. Plot the real and imaginary parts of $\kappa^* \Delta x$ vs. $\kappa \Delta x$ for $0 \leq \kappa \Delta x \leq \pi$. Derive the two leading terms in the truncation error for each scheme.

